**Kế Hoạch Phát Triển**

**Phần Mềm**

**cho**

**Ứng Dụng Ứớc Lượng Chi Phí Phần Mềm Và Thu Thập Yêu Cầu Khách Hàng**

Phiên bản 1.0 được phê chuẩn

**Được chuẩn bị bởi**

Trần Hữu Tính 1111350

Ngô Minh Phương 1111327

Đại học Cần Thơ

05/01/2015

# Mục lục

[Mục lục ii](#_Toc396491284)

[Theo dõi phiên bản tài liệu](#_Toc396491285) iii

[1 Tổ chức nhóm phát triển phần mềm 1](#_Toc396491286)

[2 Quản lý chất lượng phần mềm 1](#_Toc396491287)

[2.1 Tổ chức nhóm quản lý chất lượng 1](#_Toc396491288)

[2.2 Đề xuất các thủ tục, chuẩn, quy định về chất lượng 1](#_Toc396491289)

[2.3 Kiểm soát chất lượng 4](#_Toc396491290)

[2.4 Công cụ, kỹ thuật và phương pháp đảm bảo chất lượng 4](#_Toc396491291)

[3 Quản lý rủi ro 4](#_Toc396491292)

[4 Quản lý cấu hình 6](#_Toc396491293)

[4.1 Tổ chức nhóm quản lý cấu hình 6](#_Toc396491294)

[4.2 Nhận dạng cấu hình: 6](#_Toc396491295)

[4.3 Quản lý phát hành và phân phối 10](#_Toc396491296)

[4.4 Các tài nguyên quản lý cấu hình phần mềm 11](#_Toc396491297)

# Theo dõi phiên bản tài liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên** | **Ngày** | **Lý do thay đổi** | **Phiên bản** |
| Kế hoạch phát triển phần mềm | 05/01/2015 | Tạo mới | 1.0 |

# Tổ chức nhóm phát triển phần mềm

**Cơ cấu tổ chức**

|  |  |  |
| --- | --- | --- |
| Họ và tên | Chức vụ | Vai trò |
| Ngô Minh Phương | Thành viên | Phân tích, thiết kế, lập trình, kiểm thử, viết tài liệu. |
| Trần Hữu Tính | Thành viên | Phân tích, thiết kế, lập trình, kiểm thử, viết tài liệu. |

**Giao tiếp**

Nhóm liên lạc với nhau bằng các phương tiện như: email, điện thoại và các buổi làm việc trực tiếp tại khoa khu III.

# Quản lý chất lượng phần mềm

## Tổ chức nhóm quản lý chất lượng

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Chức vụ** | **Vai trò** |
| Trần Hữu Tính | Thành viên | Đảm bảo chất lượng tài liệu. |
| Ngô Minh Phương | Thành viên | Đảm bảo chất lượng code, kiểm thử. |

## Đề xuất các thủ tục, chuẩn, quy định về chất lượng

|  |  |
| --- | --- |
| Thành phần | Quy ước |
| Tài liệu | Quy tắc đặt tên: LV\_TenTaiLieu\_vx.x với x.x là số phiên bản, nếu có tên người tạo file thì đặt thêm hậu tố là tên người, ví du: NL04\_TenTaiLieu\_TenTacGiaVietLienKhongDau\_vx.x.  Font chữ: Arial.  Kích thước: 12px.  Màu chữ: đen.  Các bảng phải được chỉnh cho lấp đầy phần ngang không gian.  Các mục heading phải được đánh số, viết thường, hoa đầu dòng.  Cấu trúc tài liệu:   * Trang cover: Ghi tên tài liệu, tên dự án, người chuẩn bị tài liệu, người phê chuẩn, ngày chuẩn bị, số phiên bản. Font-size: 14 – 32, màu chữ: đen. * Trang nội dung:   + Trang mục lục: chữ mục lục được định dạng heading nhưng không đánh số thứ tự. Số trang được đánh bằng số la mã, viết thường.   + Trang theo dõi phiên bản tài liệu: ghi rõ ngày sửa đổi, người sửa đổi, lý do, phiên bản. Tiêu đề được định dạng heading nhưng không đánh số thứ tự. Số trang được đánh bằng số la mã, viết thường.   + Trang nội dung khác: Các đề mục được format ở dạng heading. Đánh số theo số Ả rập: 1, 2, 3…. Các mục con nhỏ hơn được đánh số theo dạng X.X. Ví dụ : 1.1, 1.2, 1.3. Số trang được đánh số bắt đầu từ 1, được đánh số Ả-rập.   Nếu là file excel thì các trang được đổi thành các sheet với nội dung và định dạng tương tự. |
| Các quy ước thiết kế | Thiết kế theo mô hình MVC   * Các file Model: đặt tên theo quy tắc đặt tên lớp. * Các file View: đặt tên theo chức năng hiển thị, ngắn gọn, theo quy tắc Camel, ví dụ: AddAgreement, ListStudent, … * Các file Controller: đặt tên theo chức năng mà nó quản lý, ngắn gọn, theo quy tắc Camel, có hậu tố là Controller.Có thể đặt tên theo tên lớp mà nó quản lý, viết ở dạng số nhiều, ví dụ: StudentsController, RegisterCoursesController, v.v |
| Các quy ước lập trình | * Tên biến: tên biến được đặt ngắn gọn, thể hiện được mục đích khi sử dụng biến, thường là danh từ, viết theo quy tắc Camel, chữ đầu tiên viết thường. Ví dụ: numberOfStudents, isVisitted,…. Có thể sử dụng tiền tố để chỉ ra kiểu của biến, ví dụ: strName, intNumberOfWays, …. * Tên hàm: tên hàm bắt buộc phải là động từ, ngắn gọn, thể hiện mục đích của hàm, viết theo quy tắc Camel, chữ đầu tiên viết thường, ví dụ: createAgreement, editAgreement, solveEquation, updateStudentInformation, …. * Tên lớp: bắt buộc là danh từ, viết theo quy tắc Camel, ngắn gọn, thể hiện đúng bản chất của lớp, viết dưới dạng số ít, ví dụ: Student, Home, Factory, … * Các dòng phải được đánh indent phù hợp để thể hiện rõ cấu trúc chương trình. |
| Các quy ước về chú thích | * Khối chú thích:   /\*  \* Here is a block comment.  \*/  Khối chú thích được sử dụng để mô tả thông tin files, phương thức, cấu trúc dữ liệu và giải thuật. Khối chú thích phải được đặt ở đầu file, đầu các phương thức, cấu trúc dữ liệu, giải thuật, …Khối chú thích phải ghi rõ tên tác giả, nội dung chú thích, đối với phương thức phải ghi rõ đầu vào, đầu ra, phục vụ cho chức năng nào trong hệ thống.   * Chú thích dòng:   if (condition) {  // Handle the condition.  ...  }  Chú thích dòng được sử dụng để phân mức cho code, trước chú thích dòng nên là 1 khoảng trắng. Nếu số lượng chữ quá dài để viết chú thích dòng thì nên sử dụng khối chú thích.   * Chú thích theo sau:   if (a == 2) {  return TRUE; // special case  } else {  return isPrime(a); // works only for odd a  }  Chú thích này được sử dụng để chú thích cho các dòng code, phải viết ngắn gọn, khoảng cách đủ xa để phân biệt với đoạn code.   * Doc comment:   /\*\*  \* The Example class provides ...  \*/  public class Example {  ...  Chú thích dạng này dùng để chú thích lớp, giao diện, phương thức, được đặt trong /\*\*…\*/ Ghi rõ chức năng của lớp, đầu vào, đầu ra, nội dung xử lý của phương thức, v.v |
| Chuẩn kiểm thử | * Kiểm thử theo mô hình chữ V, qua 4 giai đoạn. |
| Quy ước về upload file lên Github | * Mỗi ngày, phải đồng bộ tất cả các file trên Github về máy tránh tình trạng chỉnh sửa trên phiên bản cũ. * Khi upload phải xem có xảy ra đụng độ hay không, nếu xảy ra đụng độ thì phải discard change, tải phiên bản mới nhất trên Github về rồi chỉnh sửa. * Khi upload phải ghi rõ thao tác gì được làm, chỉnh sửa file nào, trong mục Summary. Ví dụ : Upload file LV\_TenTaiLieu\_vx.x,.Mục description ghi rõ nội dung bị ảnh hưởng. * Cuối ngày, thành viên phải gửi mail báo cáo cho nhóm trưởng để nhóm trưởng nắm tình hình tiến độ làm việc của nhóm. * Khi upload source code lên Github, chỉ up thư mục src trong project. |
| Deffect Logging | * Trong quá trình làm, nếu gặp vấn đề chưa rõ, thì phải đặt Q&A trên Github, hoặc gửi email cho thành viên trong nhóm, nội dung Q&A ghi rõ nguồn, phần chưa hiểu, giải pháp của bản thân. Trong thời gian chờ đợi thì có thể làm theo giải pháp của bản thân tự đưa ra. * Nếu gặp các lỗi, phải được log lại trên mục Issuse của Github. Các thành viên có trách nhiệm lên xem phần nào có liên quan thì sửa lại và phản hồi trên Github. |

## Kiểm soát chất lượng

|  |  |
| --- | --- |
| Tài liệu | Yêu cầu |
| Software Requirement Specification | * Mô tả hệ thống. Có thể có sơ đồ chức năng của tổ chức. * Sơ đồ usecase. * Đặc tả usecase: Tên usecase, chức năng usecase, người sử dụng, tiền điều kiện, yêu cầu chức năng. * Yêu cầu phi chức năng. |
| High Level Design | * Sơ đồ usecase: phải thể hiện được chức năng của hệ thống, phân rã chi tiết đủ để thể hiện được các chức năng. * Sơ đồ dòng dữ liệu: thể hiện được các luồng dữ liệu vào ra trong hệ thống, phân rã vừa đủ để thể hiện rõ ràng được các chức năng * Cơ sở dữ liệu phải được vẽ bằng hình ảnh, thể hiện được các cột, trường, kiểu dữ liệu, mối quan hệ, ràng buộc giữa các bảng. |
| Detail Design Document | * Sơ đồ lớp và sơ đồ trình tự phải được vẽ bằng hình ảnh. * Sơ đồ lớp: Các lớp phải được chia thành các pakage. * Sơ đồ trình tự phải thể hiện ít nhất 1 trường hợp cho mỗi chức năng trong hệ thống. |
| Testing document | * Các trường hợp kiểm thử, yêu cầu các trường hợp kiểm thử phải có độ bao phủ trên 90%. |

## Công cụ, kỹ thuật và phương pháp đảm bảo chất lượng

Sử dụng các công cụ:

* CheckStyle: dùng để kiểm tra nội dung code có đạt chuẩn quy tắc code.
* FindBugs: dùng để tìm các bug trong file Java như bắt các trường hợp có thể sinh ra exception.
* Junit: kiểm thử các hàm trong code.
* Quick Test Pro: kiểm thử tích hợp, giao diện, v.v

# Quản lý rủi ro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Yếu tố rủi ro** | **Mức độ rủi ro** | **Chiến lược làm giảm rủi ro** | **Hướng giải quyết khi xảy ra rủi ro** |
| Nhóm yêu tố rủi ro liên quan đến khách hàng và người sử dụng | | | |  |
| 1 | Khách hàng thay đổi yêu cầu, chức năng của sản phẩm. | Cao | Phỏng vấn cẩn thận từng yêu cầu của khách hàng đối với hệ thống. Đưa ra các gợi ý về những chức năng cần thiết cho khách hàng chọn. | Họp nhóm phân tích lại yêu cầu của khách hàng để hoàn thiện phần mềm. |
| 2 | Người dùng không hiểu rõ về hệ thống cũng như các chức năng của nó. | Trung bình | Cần có mục hướng dẫn để giúp khách hàng dễ dàng trong việc sử dụng. | Tổ chức buổi hướng dẫn khách hàng sử dụng chương trình. |
| Nhóm yếu tố liên quan đến phạm vi và các yêu cầu. | | | |  |
| 1 | Thay đổi, mở rộng phạm vi dự án. | Thấp | Phỏng vấn kỹ lưỡng về quy mô và nhu cầu mở rộng quy mô của khách hàng. | Họp nhóm, dự trù lại kinh phí, phân tích lại yêu cầu để hoàn thiện tài liệu. |
| 2 | Yêu cầu không rõ ràng gây ra sự hiểu sai. | Trung bình | Phỏng vấn khách hàng một cách tỉ mỉ về các yêu cầu của họ để tránh hiểu sai ý khách hàng. | Thay đổi ngay và lập tức các yêu cầu sai lệch. Thiết kế lại chức năng tương ứng, kiểm tra xem nó có phù hợp với các chức năng khác hay không? |
| Nhóm yếu tố liên quan đến sự thực hiện. | | | |  |
| 1 | Không đồng nhất trong việc đặt tên biến cũng như tên hàm… | Trung bình | Cần có quy ước đặt tên hàm, biến do nhóm quy định(có thể tham khảo các chuẩn đặt tên ….) | Cần tập trung nhóm lại thay đổi và thống nhất triệt để các vấn đề trong việc lập trình. |
| 2 | Thiếu kinh phí | Trung bình | Cần dự trù kinh phí và chi phí phát sinh trong quá trình phát triển tránh trường hợp vượt kinh phí dự kiến. | Vận động kinh phí từ phía các đối tác. |
| 3 | Chậm trễ trong tiến độ do vấn đề cá nhân( về quê, bệnh…..) | Thấp | Các thành viên ở xa có thể làm việc bằng hình thức online. | Nhóm trưởng xúc tiến hoặc phân chia lại công việc nhầm tăng tiến độ các thành viên để kế hoạch hoàn thành đúng thời hạn. |
| 4 | Phản ứng chậm trể các vấn đề đã thông báo. | Trung bình | Thông tin phản hồi phải được thông báo trong một thời gian nhất định kể từ khi thông báo. | Liên hệ trực tiếp đến thành viên đó để thông báo và phê bình. |
| Nhóm yếu tố liên quan đến môi trường. | | | |  |
| 1 | Các thành viên ở xa dẫn đến không thuận lợi trong việc đi lại cũng như họp nhóm làm ảnh hưởng đến tiến độ công việc. | Thấp | Cần lên kế hoạch và nội dung họp nhóm một cách cụ thể. Các thành viên trong nhóm cần phải chủ động thời gian. | Họp nhóm phân chia lại công việc đáp ứng tiến độ của kế hoạch. |
| 2 | Phần mềm không tương thích khi thay đổi môi trường hoạt động( phần cứng, phần mềm, hệ điều hành). | Trung bình | Khi phỏng vấn cần chú ý yếu tố môi trường hoạt động của phần mềm mà khách hàng sử dụng (phần cứng, hệ điều hành). | Tiến hành cải tiến phần mềm để phù hợp với môi trường làm việc mới. |

# Quản lý cấu hình

## 4.1 Tổ chức nhóm quản lý cấu hình

|  |  |  |
| --- | --- | --- |
| Họ và tên | Chức vụ | Vai trò |
| Trần Hữu Tính | Thành viên | Quản lý cầu hình của: Tài liệu đặc tả yêu cầu, thiết kế yêu cầu, phân tích yêu cầu… |
| Ngô Minh Phương | Thành viên | Quản lý cầu hình của: Thiết kế cơ sở dữ liệu, High level design, Detail design… |

- **Quy tắc chung:**

Sử dụng GitHub để quản lý cấu hình, các thành viên trong nhóm phải thông qua lịch biểu để update tài liệu đúng theo thời gian. Không được xóa bất kỳ tài liệu nào làm ảnh hưởng tới công việc của nhóm.

Công việc của thành viên nào thì thao tác trên tài liệu trong thư mục của thành viên đó.

**- Tổ chức lưu trữ**

/LuanVan/ thư mục chính

|----docs/ Lưu trữ các tài liệu trong công việc.

| |----plan/ Lưu trữ lịch phân công công việc.

| |----design/ Lưu trữ các tài liệu thiết kế.

| |----user guide/ Lưu trữ tài liệu hướng dẫn sử dụng.

|--src/ Lưu trữ source code.

|--database/ Lưu trữ cơ sở dữ liệu.

|--test/ Lưu trữ các tài liệu kiểm thử.

## 4.2 Nhận dạng cấu hình:

4.2.1 High Level Design

- Tên: High Level Design

- Cấu trúc thư mục: LuanVan/docs/design/LV\_HighLevelDesign.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.2 Tài liệu hướng dẫn sử dụng phần mềm

- Tên: Tài liệu hướng dẫn sử dụng phần mềm (User guide)

- Cấu trúc thư mục: LuanVan/docs/user guide/LV\_UserGuide.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.3 Bản đặc tả yêu cầu

- Tên: Bản đặc tả yêu cầu (Software Requirements Specification)

- Cấu trúc thư mục: LuanVan/docs/design/LV\_SoftwareRequirementsSpecification.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.4 Bản thiết kế giao diện

- Tên: Bản thiết kế giao diện (User Interface Design)

- Cấu trúc thư mục: LuanVan/docs/design/LV\_UserInterfaceDesign.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx, .xlsx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.5 Bản thiết kế cơ sở dữ liệu

- Tên: Bản thiết kế cơ sở dữ liệu (Database Design)

- Cấu trúc thư mục: LuanVan/docs/design /LV\_DatabaseDesign .docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx, .cdm
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.6 Lịch phân công

- Tên: Lịch phân công (Schedule)

- Cấu trúc thư mục: LuanVan/docs/plan/LV\_Schedule.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .xlsx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.7 Source Code

- Tên: Source Code

- Cấu trúc thư mục: LuanVan/source/

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .java
* Ngôn ngữ: JAVA
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.8 Database

- Tên: Database

- Cấu trúc thư mục: LuanVan/database/

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .sql
* Ngôn ngữ: MySQL
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.9 Detail Design

- Tên: Detail Design

- Cấu trúc thư mục: LuanVan/docs/design/LV\_DetailDesign.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .docx, .xlsx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

4.2.10 Tài liệu kiểm thử

Tên: Tài liệu kiểm thử (Unit Testing)

- Cấu trúc thư mục: LuanVan/test/LV\_UnitTesting.docx

- Thuộc tính:

* Tác giả: Ngô Minh Phương, Trần Hữu Tính.
* Kiểu file: .xlsx
* Ngôn ngữ: Tiếng Việt.
* Những người chịu trách nhiệm: Ngô Minh Phương, Trần Hữu Tính.

## 4.3 Quản lý phát hành và phân phối

Trước khi phát hành sản phẩm và phân phối, người chịu trách nhiệm phát hành chuẩn bị bản mô tả thành phần sản phẩm phải cung cấp những thông tin cần thiết về sản phẩm được phát hành cho những người liên quan. Thông tin đó bao gồm: ngày phát hành, tên sản phẩm, mục đích của đợt phát hành, hướng dẫn sử dụng, và các thông tin khác như giới thiệu tóm tắt về sản phẩm, dấu hiệu nhận biết sản phẩm, những điều cần chú ý, nếu cần.

Các biểu mẫu được sử dụng cho việc bàn giao, phát hành sản phẩm là:

* Biên bản bàn giao tài liệu.
* Biên bản bàn giao phần mềm.
* Thông báo phát hành.

Việc sử dụng các biểu mẫu này sẽ do quản trị dự án quyết định.

**Chi tiết kế hoạch thực hiện:**

*(Tuần 1 bắt đầu từ ngày 04/08.2014)*

|  |  |  |  |
| --- | --- | --- | --- |
| Tuần bắt đầu - kết thúc | Tên công việc | Tên tài liệu | Trạng thái |
| 3 - 3 | - Lập lịch phân công  - Lập kế hoạch thực hiện | - Lịch phân công  - Kế hoạch thực hiện | Hoàn thành |
| 4 - 5 | - Lập bản đặc tả yêu cầu phần mềm.  - Lập high level design:  +Mô tả hệ thống.  +Mô tả chức năng. | - Bản đặc tả yêu cầu phần mềm.  -High level design:  +Use case.  +CDM. | Hoàn thành |
| 6 - 7 | - Viết Detail design:  +Sơ đồ dòng dữ liệu.  +Mã giả.  +Activity diagram.  +Thiết kế giao diện.  +Cơ sở dữ liệu. | Detail design document | Hoàn thành |
| 8 - 10 | - Tạo database  - Lập trình | - Database  - Source code | Hoàn thành, riêng phần source code có thể chưa hoàn thành (nhưng ít nhất phải >80%) |
| 11 - 12 | - Lập tài liệu kiểm thử  - Tạo tài liệu hướng dẫn sử dụng | - Tài liệu kiểm thử  - Tài liệu hướng dẫn sử dụng | Hoàn thành |
| 13 - 13 | - Chuẩn bị cho việc báo cáo | -Báo cáo, bàn giao | Hoàn thành |

Bảng 4.3.1: Chi tiết thực hiện kế hoách

## 4.4 Các tài nguyên quản lý cấu hình phần mềm

**- Phần cứng**

* **Bộ xử lý:** 1 gigahertz (GHz) trở lên
* **RAM:** 1 gigabyte (GB) (32 bit) hoặc 2 GB (64 bit)
* **Dung lượng ổ cứng:** 5 GB (32 bit) hoặc 7 GB (64 bit)

**- Phần mềm**

4.4.1 GitHub

* Tên: GitHub
* Loại: Công cụ.
* Mục đích sử dụng: Quản lý cấu hình.
* Vai trò sử dụng: Các thành viên trong nhóm.

4.4.2 Microsoft Office

* Tên: Microsoft Office
* Thuộc tính: Phiên bản Microsoft Office từ 2010 trở lên
* Loại: Công cụ.
* Mục đích sử dụng: Tạo tài liệu.
* Vai trò sử dụng: Các thành viên trong nhóm.

4.4.3 Power Designer

* Tên: Power Designer
* Thuộc tính: Phiên bản Power Designer 16 trở lên
* Loại: Công cụ.
* Mục đích sử dụng: Thiết kế sơ đồ, thiết kế cơ sở dữ liệu.
* Vai trò sử dụng: Các thành viên trong nhóm.

4.4.4 Eclipse

* Tên: Eclipse
* Thuộc tính: Phiên bản Eclipse 3.0 trở lên
* Loại: Công cụ.
* Mục đích sử dụng: Lập trình.
* Vai trò sử dụng: Các thành viên trong nhóm.